IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Mohammed Alsuwaie
May 04, 2024

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion
- Appendix

Executive Summary

- **Summary of methodologies:**
  - Collected Falcon 9 historical launch records from SpaceX API and SpaceX Wikipedia page.
  - Created a 'class' column to classify successful landings.
  - Explored data using SQL, visualization tools, folium maps, and dashboards.
  - Selected relevant columns as features for modeling.
  - Converted all categorical variables to binary format using one hot encoding.
  - Standardized data and optimized machine learning models using GridSearchCV.
  - Visualized the accuracy scores of all models.
  - Machine Learning Model Prediction.
- **Summary of all results:**
  - Developed four machine learning models: Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K Nearest Neighbors.
  - All models showed similar performance with an accuracy rate of about 83.33%.
  - The models exhibited high recall, yet their precision and specificity varied, indicating proficiency in identifying successful launches while highlighting challenges in accurately predicting unsuccessful ones.

Introduction

- **Project background and context**
  - SpaceX significantly undercuts traditional rocket launch costs, advertising Falcon 9 launches at $62 million compared to competitors' prices of around $165 million.
  - Much of SpaceX's cost advantage stems from its ability to reuse the first stage of the rocket. Therefore, predicting the success of these first-stage landings is crucial for cost estimation.
  - The main objective is to develop a machine-learning pipeline to predict whether the first stage of a rocket will land successfully. This predictive capability could be useful for competitive bidding situations against SpaceX.
- **Problems you want to find answers**
  - What factors are most predictive of a successful rocket landing?
  - How do various features interact to determine the likelihood of a successful landing?
  - What operating conditions to maintain a successful landing program,

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Collected Falcon 9 historical launch records from SpaceX API and SpaceX Wikipedia page titled List of Falcon 9 and Falcon Heavy launches.
- Perform data wrangling
  - Converted all categorical variables to binary format using one hot encoding.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- **How data sets were collected:**
  - Make a get request to the SpaceX API and clean the requested data.
  - Web scrap Falcon 9 launch records with (BeautifulSoup) to extract a Falcon 9 launch records HTML table from Wikipedia.
  - Parse the table and convert it into a Pandas data frame.
- **SpaceX API Data Columns:**
  - FlightNumber-Date-BoosterVersion-PayloadMass-Orbit-LaunchSite-Outcome-Flights-GridFins-Reused-Legs-LandingPad-Block-ReusedCount-Serial-Longitude-Latitude).
- **Wikipedia Web Scraping Data Columns:**
  - Flight No.-Launch site-Payload-PayloadMass-Orbit-Customer-Launch outcome-Version Booster-Booster landing-Date-Time

Data Collection – SpaceX API

- **Data collection with SpaceX**
- **GitHub URL:**

[Applied-Data-Science-Capstone/Data Collection-SpaceX API.ipynb at main · msuwaie/Applied-Data-Science-Capstone (github.com)]

1. Retrieve and process the SpaceX launch data using an HTTP GET request.
2. convert the response content to JSON using the method and transform it into a Pandas Data Frame using json_normalize().
3. Using “BoosterVersion” to narrow down the Data Frame to include only the Falcon 9 launches data.
4. Preform data wrangling to deal with missing values in the dataset

Data Collection - Scraping

- **Web scraping process**
- **GitHub URL:**

[Applied-Data-Science-Capstone/Data Collection-Web Scraping.ipynb at main · msuwaie/Applied-Data-Science-Capstone (github.com)]

1. Retrieve historical launch data for Falcon 9 from the HTML table on the Wikipedia page titled "List of Falcon 9 and Falcon Heavy launches."
2. Execute an HTTP GET request to obtain the Falcon 9 Launch HTML page as an HTTP response.
3. Extract all the column and variable names from the header of the HTML table.
4. Create a Data Frame by parsing the launch data from the HTML tables

Data Wrangling

- **How data were processed:**
- Conducted Exploratory Data Analysis (EDA) to identify patterns in the dataset and establish the label for training supervised models.
- The dataset includes various instances of unsuccessful booster landings. For example:
    - "True Ocean" indicates a successful ocean landing, while "False Ocean" signifies an unsuccessful one.
    - "True RTLS" denotes a successful return to the launch site (RTLS) landing, and "False RTLS" indicates a failure.
    - "True ASDS" refers to a successful drone ship landing, whereas "False ASDS" means the landing was unsuccessful.
- Convert these outcomes into training labels, where '1' represents a successful landing and '0' indicates an unsuccessful landing.
- **GitHub URL:**

[Applied-Data-Science-Capstone/Spacex-Data Wrangling.ipynb at main · msuwaie/Applied-Data-Science-Capstone (github.com)]

1. Determine the total number of launches at each site.
2. Determine the frequency of each orbit type.
3. Compute the frequency and types of mission outcomes for each orbit.
4. Generate a landing outcome label from the Outcome column.

EDA with Data Visualization

- Summary of the charts that were created:
    - “Catplot” plot to visualize the relationship between Flight Number and Launch Site.
    - “Scatter” plot to visualize the relationship between Payload Mass and Launch Site.
    - “Bar Chart” to visualize the relationship between the success rate of each orbit type.
    - “Catplot” plot to visualize the relationship between Flight Number and Orbit type.
    - “Catplot” plot to visualize the relationship between Payload and Orbit type.
    - “Line Chart” plot to visualize the launch success yearly trend.
- **Reasons for using those specific charts:**
    - Gaining some initial insights into how key variables might influence the success rate. Based on these insights, we will select the features that will be used for predicting success in future modules.
- **GitHub URL:**

[Applied-Data-Science-Capstone/EDA Data Visualization.ipynb at main · msuwaie/Applied-Data-Science-Capstone (github.com)]

EDA with SQL

- **Summary of the SQL queries executed:**
  - Connect to a SQLite database
  - Import data into an SQLite database.
  - Execute a query to identify all unique launch sites.
  - Execute a query to locate records where launch sites start with 'CCA'.
  - Run a query to calculate the total payload mass carried by boosters on missions launched by NASA (CRS).
  - The aggregate count of missions that were successful and those that failed.
  - The unsuccessful drone ship landing outcomes, along with the corresponding booster versions and names of the launch sites
- **GitHub URL:**

Applied-Data-Science-Capstone/EDA with SQL.ipynb at main · msuwaie/Applied-Data-Science-Capstone (github.com)

Build an Interactive Map with Folium

- **A summary of the map objects such as markers, circles, lines, etc., that have been created and added to a Folium map:**
    - Add markers for each site to the map using the site's latitude and longitude coordinates.
    - Create a Folium map object centered initially on the NASA Johnson Space Center in Houston, Texas.
    - Use “folium.Circle” to add a highlighted circle area with a text label at a specific coordinate on the map.
    - Add folium.Circle and folium.Marker for each launch site on the map, highlighting the location of each site.
- **The purpose of adding those specific objects:**
    - Quickly locate the Easily determine which launch sites have relatively high success rates.
    - coordinates of any points of interest, such as railways, highways, coastlines, etc.
- **GitHub URL:**

[Applied-Data-Science-Capstone/Interactive Map with Folium.ipynb at main · msuwaie/Applied-Data-Science-Capstone (github.com)]

Build a Dashboard with Plotly Dash

- **Summary of the plots, graphs, and interactive elements that have been incorporated into the dashboard:**

Built an interactive dashboard with Ploty dash, includes both a pie chart and a scatter plot for visual analysis:

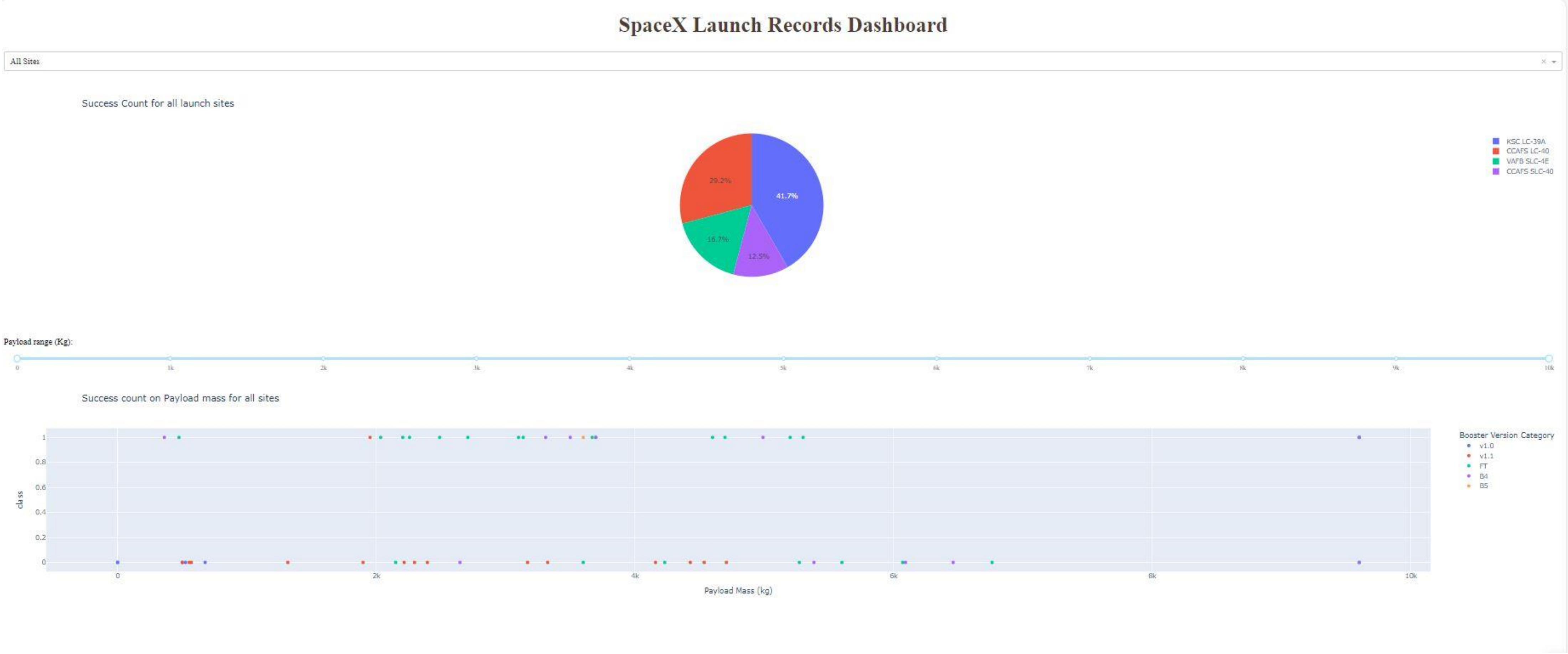
  - The pie chart displays the distribution of successful landings. It can be configured to show success rates across all launch sites or for each site.
  - The scatter plot allows for a dynamic analysis based on the selected criteria: either an overall view of all sites or a detailed view of a single site, with payload mass adjustable via a slider ranging from 0 to 10,000 kg.

- **The purpose behind including those specific plots and interactive features:**
  - The pie chart specifically helps in visualizing the success rate at different launch sites, making it easier to compare performance across sites.
  - The scatter plot is useful for visualizing how success rates vary depending on launch sites, payload mass, and the category of booster version

- **GitHub URL (SpaceX python file):**

Applied-Data-Science-Capstone/spaceX_dash_app.py at main · msuwaie/Applied-Data-Science-Capstone (github.com)

SpaceX Launch Dashboard with Plotly Dash



Predictive Analysis (Classification)

- The process of building, evaluating, improving, and finding the best-performing classification model:
  - Split the data into training and testing data using the function (train_test_split).
  - Create a logistic regression object then create a GridSearchCV object logreg_cv with cv = 10.
  - Calculate the accuracy of the test data using the method score.
  - Create a decision tree classifier object then create a GridSearchCV object tree_cv with cv = 10.
  - Create a k nearest neighbors object then create a object with cv = 10.
  - Calculate the accuracy of knn_cv on the test data using the method.
  - Creating the bar plot to find the method that performs best.
- GitHub URL:

[Applied-Data-Science-Capstone/SpaceX_Machine Learning Prediction.ipynb at main · msuwaie/Applied-Data-Science-Capstone (github.com)]

Split the data into training and testing data using the function (train_test_split).

1

Create a logistic regression object then create a GridSearchCV object logreg_cv with cv = 10.

2

Calculate the accuracy of the test data using the method score.

3

Create a decision tree classifier object then create a GridSearchCV object tree_cv with cv = 10.

4

Create a k nearest neighbors object then create a object with cv = 10.

5

Calculate the accuracy of knn_cv on the test data using the method.

6

Creating the bar plot to find the method performs best.

7

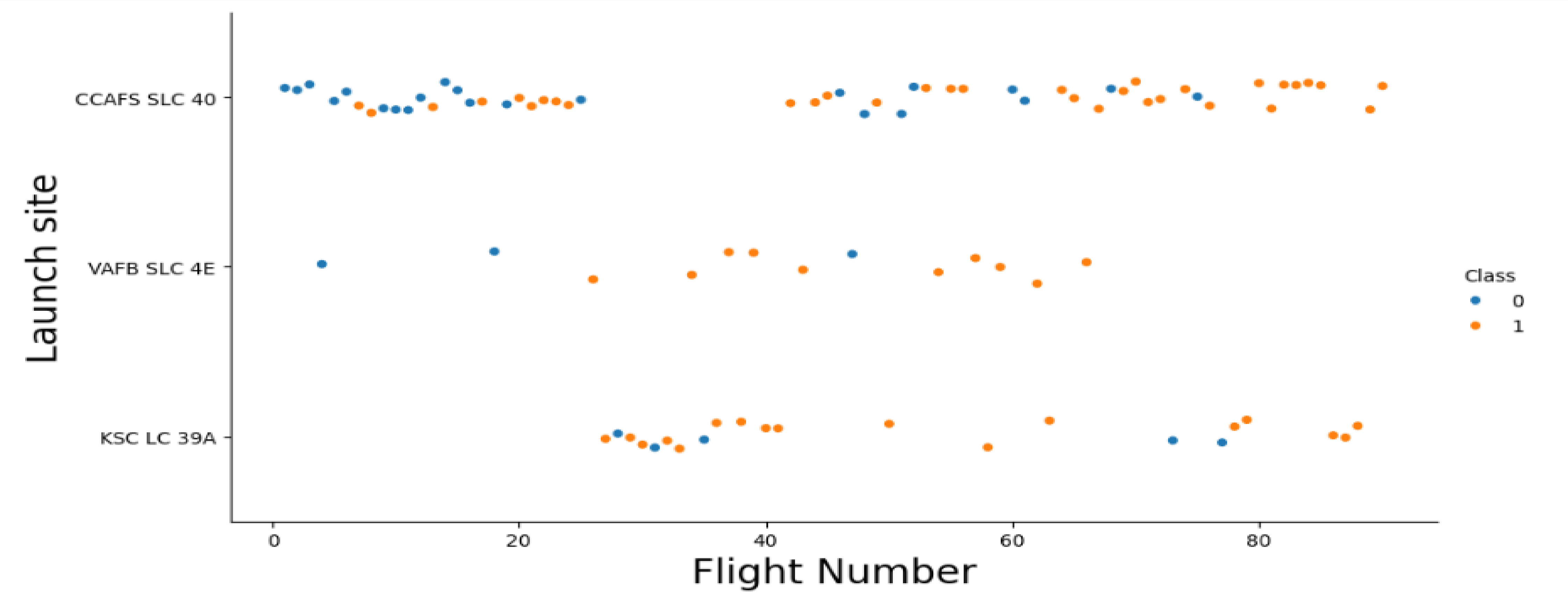
Results

- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

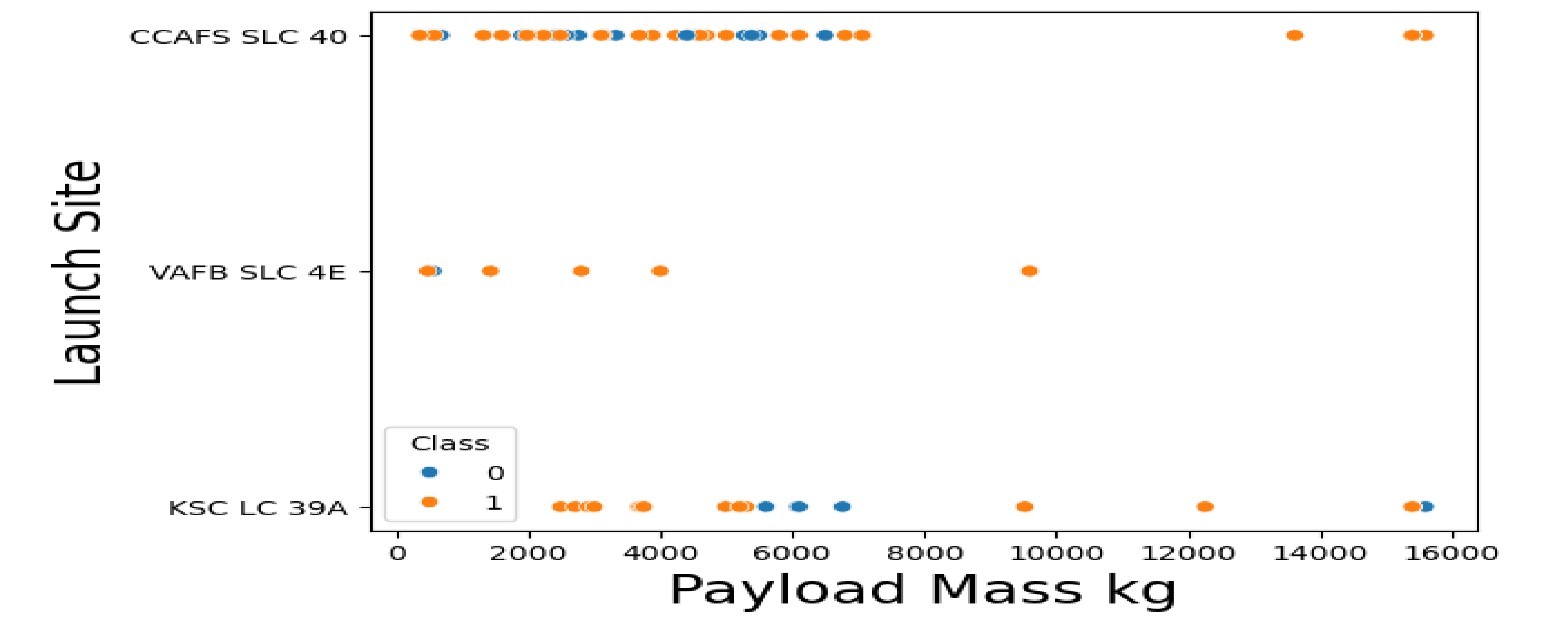
Insights drawn from EDA

Flight Number vs. Launch Site



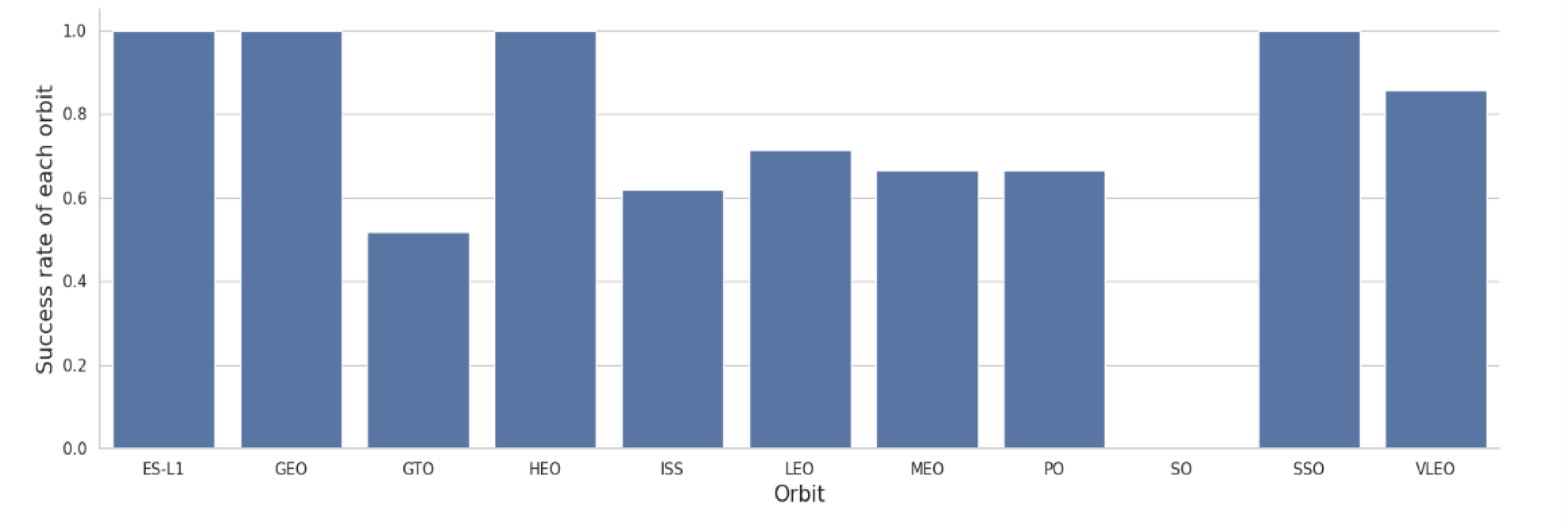
Scatter point plots (Orange (1)) indicate a successful launch while scatter point plots (Blue (0)) indicate an unsuccessful launch. As the number of flights conducted at a launch site increases, so does the success rate

Payload vs. Launch Site



Scatter point plots (Orange (1)) indicate a successful launch while scatter point plots (Blue (0)) indicate an unsuccessful launch. The higher the payload mass for the launch site CCAFS SLC 40, the greater the success rate

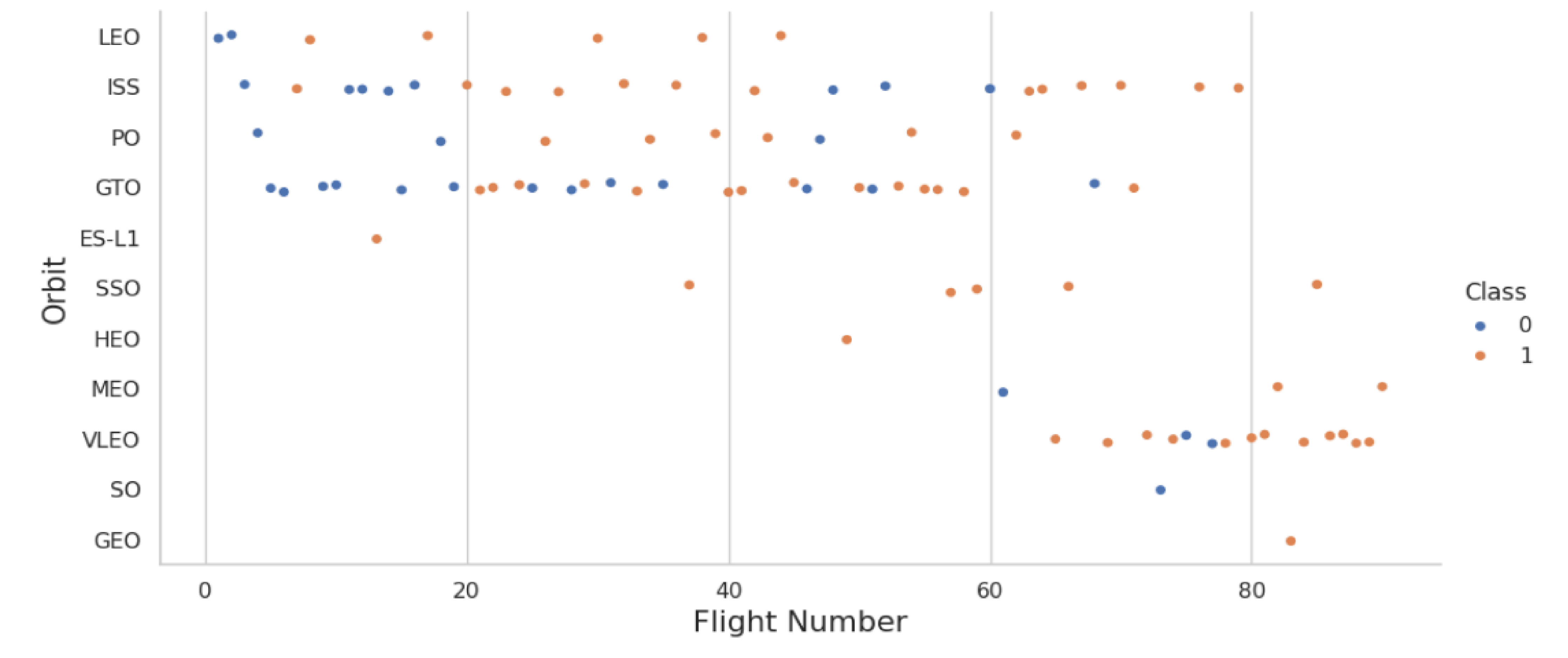
Success Rate vs. Orbit Type



Orbits have the highest success rate of 1.0:

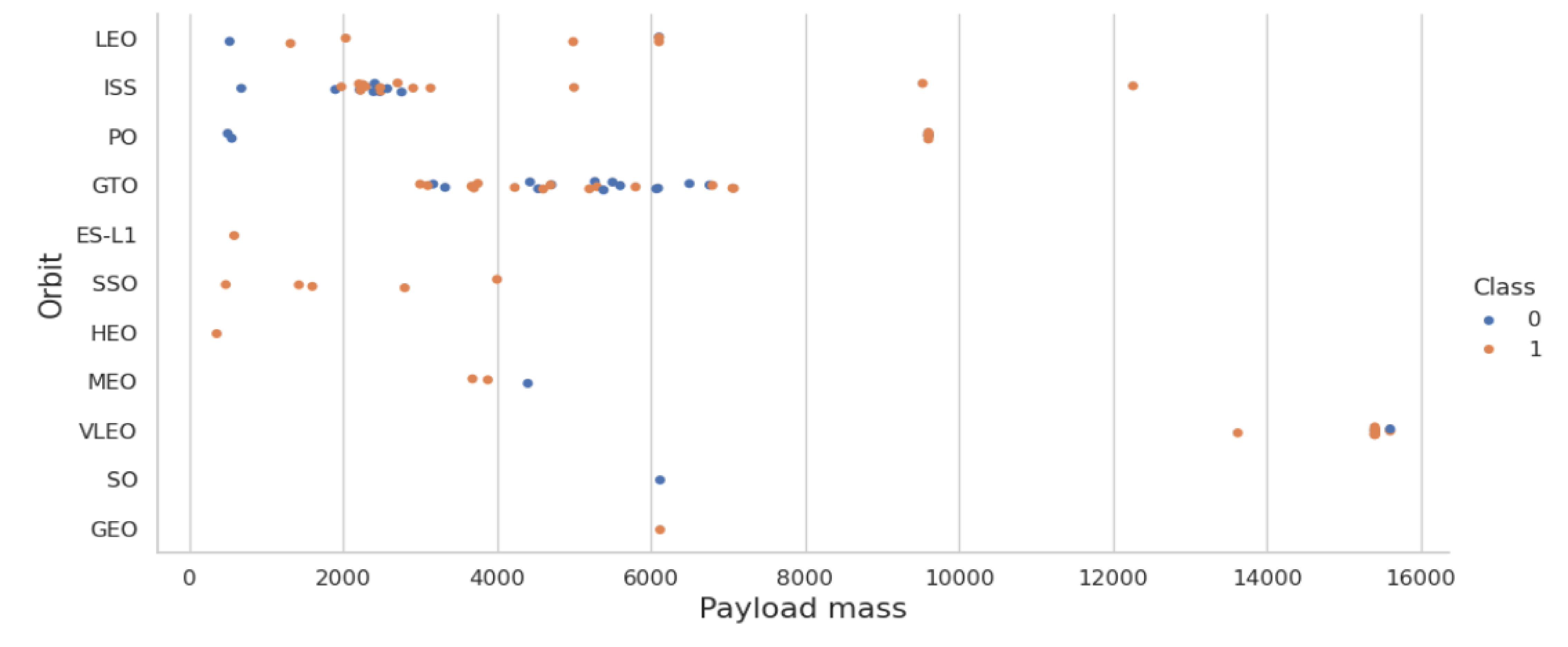
1. ES-L1
2. GEO
3. HEO
4. SSO

Flight Number vs. Orbit Type



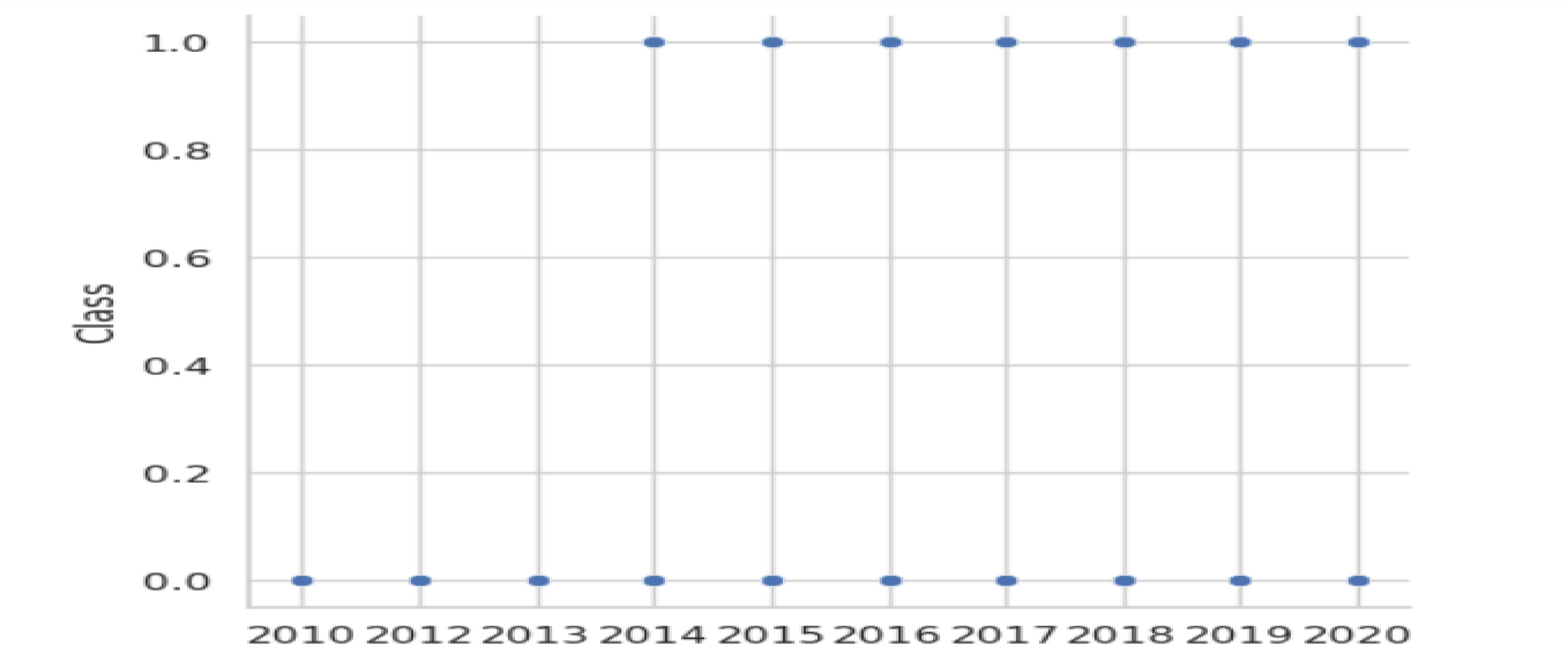
In Low Earth Orbit (LEO), success appears to be correlated with the number of flights, while in Geostationary Transfer Orbit (GTO), there is no discernible relationship between flight frequency and orbital outcomes.

Payload vs. Orbit Type



Successful landings are more frequent for Polar (PO), Low Earth Orbit (LEO), and International Space Station (ISS) orbits when dealing with heavy payloads.

Launch Success Yearly Trend



The success rate since 2013 kept increasing till 2020.

All Launch Site Names

- The names of the unique launch sites:

```
In [10]: %sql SELECT DISTINCT LAUNCH_SITE as "Launch_Sites" FROM SPACEXTBL;

 * sqlite:///my_data1.db
Done.
```

Out[10]:

Launch_Sites
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

“DISTINCT” keyword is used to show only unique launch sites from the SpaceX data.

Launch Site Names Begin with 'CCA'

Show 5 records where launch sites begin with `CCA`:

In [11]:
```
%sql SELECT * FROM 'SPACEXTBL' WHERE Launch_Site LIKE 'CCA%' LIMIT 5;
```

```
 * sqlite:///my_data1.db
Done.
```

Out[11]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

The total payload carried by boosters from NASA:

```
In [13]:
# Execute query to find the total payload mass carried by boosters launched by NASA (CRS)
%sql SELECT SUM(PAYLOAD_MASS__KG_) as "Total Payload Mass(Kgs)", Customer FROM 'SPACEXTBL' WHERE Customer = 'NASA (CRS)';

 * sqlite:///my_data1.db
Done.
```

Out[13]:

Total Payload Mass(Kgs)	Customer
45596	NASA (CRS)

Average Payload Mass by F9 v1.1

The average payload mass carried by booster version F9 v1:

```
In [15]: %sql SELECT AVG(PAYLOAD_MASS__KG_) as "Payload Mass Kgs", Customer, Booster_Version FROM 'SPACEXTBL' WHERE Booster_Version l

 * sqlite:///my_data1.db
Done.
```

Out[15]:

Payload Mass Kgs	Customer	Booster_Version
2534.6666666666665	MDA	F9 v1.1 B1003

The average payload mass of F9 1.1 is on the low end of the payload mass range.

First Successful Ground Landing Date

The date of the first successful landing outcome on the ground pad was achieved:

```
[44]: %sql SELECT MIN(DATE) FROM 'SPACEXTBL' WHERE "Landing_Outcome" = "Success (ground pad)";

 * sqlite:///my_data1.db
Done.
```

[44]:

MIN(DATE)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

List of the names of boosters that have successfully landed on drone ships and had payload mass greater than 4000 but less than 6000:

In [42]:
```
%sql SELECT DISTINCT Booster_Version, Payload FROM SPACEXTBL WHERE "Landing_Outcome" = "Success (drone ship)" AND PAYLOAD_M/
```

```
 * sqlite:///my_data1.db
Done.
```

Out[42]:

Booster_Version	Payload
F9 FT B1022	JCSAT-14
F9 FT B1026	JCSAT-16
F9 FT B1021.2	SES-10
F9 FT B1031.2	SES-11 / EchoStar 105

Four boosters that have successful drone ship landings with payload mass between 4000 and 6000.

Total Number of Successful and Failure Mission Outcomes

The total number of successful and failed mission outcomes:

In [24]:
```
%sql SELECT "Mission_Outcome", COUNT("Mission_Outcome") as Total FROM SPACEXTBL GROUP BY "Mission_Outcome";
```

```
 * sqlite:///my_data1.db
Done.
```

Out[24]:

Mission_Outcome	Total
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

The names of the boosters that have carried the maximum payload mass:

```
In [26]:  # Find the maximum payload mass
          %sql SELECT "Booster_Version",Payload, "PAYLOAD_MASS__KG_" FROM SPACEXTBL WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_M
```

```
 * sqlite:///my_data1.db
Done.
```

Out[26]:

Booster_Version	Payload	PAYLOAD_MASS__KG_
F9 B5 B1048.4	Starlink 1 v1.0, SpaceX CRS-19	15600
F9 B5 B1049.4	Starlink 2 v1.0, Crew Dragon in-flight abort test	15600
F9 B5 B1051.3	Starlink 3 v1.0, Starlink 4 v1.0	15600
F9 B5 B1056.4	Starlink 4 v1.0, SpaceX CRS-20	15600
F9 B5 B1048.5	Starlink 5 v1.0, Starlink 6 v1.0	15600
F9 B5 B1051.4	Starlink 6 v1.0, Crew Dragon Demo-2	15600
F9 B5 B1049.5	Starlink 7 v1.0, Starlink 8 v1.0	15600
F9 B5 B1060.2	Starlink 11 v1.0, Starlink 12 v1.0	15600
F9 B5 B1058.3	Starlink 12 v1.0, Starlink 13 v1.0	15600
F9 B5 B1051.6	Starlink 13 v1.0, Starlink 14 v1.0	15600
F9 B5 B1060.3	Starlink 14 v1.0, GPS III-04	15600
F9 B5 B1049.7	Starlink 15 v1.0, SpaceX CRS-21	15600

2015 Launch Records

The failed landing outcomes in drone ships, their booster versions, and launch site names for in year 2015:

In [20]:

```
# Convert the 'Date' column to datetime to facilitate extraction of year and month
df['Date'] = pd.to_datetime(df['Date'])

# Filter for year 2015 and failure landing outcomes in drone ship
filtered_df = df[(df['Date'].dt.year == 2015) &
                 (df['Landing_Outcome'] == 'Failure (drone ship)')].copy()

# Extract the month as a new column
filtered_df['Month'] = filtered_df['Date'].dt.month_name()

# Select the required columns
final_df = filtered_df[['Month', 'Booster_Version', 'Launch_Site', 'Landing_Outcome']]

print(final_df)
```

```
      Month Booster_Version  Launch_Site       Landing_Outcome
13  January    F9 v1.1 B1012  CCAFS LC-40  Failure (drone ship)
16    April    F9 v1.1 B1015  CCAFS LC-40  Failure (drone ship)
```

Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order:

In [43]:

```
%sql SELECT * FROM SPACEXTBL WHERE "Landing_Outcome" LIKE 'Success%' AND (Date BETWEEN '2010-06-04' AND '2017-03-20') ORDER
```

```
 * sqlite:///my_data1.db
Done.
```

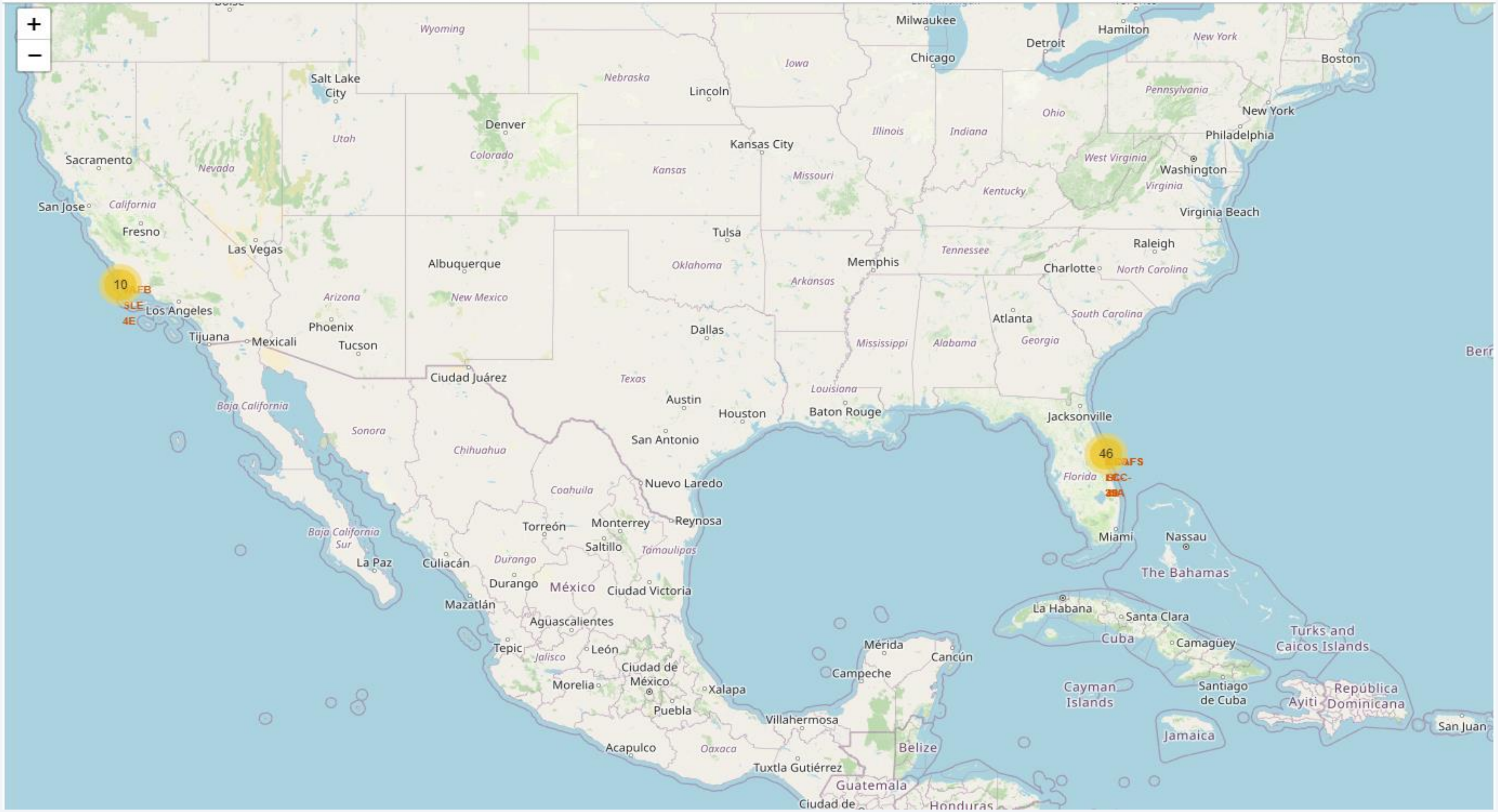
Out[43]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outc
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (gro l
2017-01-14	17:54:00	F9 FT B1029.1	VAFB SLC-4E	Iridium NEXT 1	9600	Polar LEO	Iridium Communications	Success	Success (dr s
2016-08-14	5:26:00	F9 FT B1026	CCAFS LC-40	JCSAT-16	4600	GTO	SKY Perfect JSAT Group	Success	Success (dr s
2016-07-18	4:45:00	F9 FT B1025.1	CCAFS LC-40	SpaceX CRS-9	2257	LEO (ISS)	NASA (CRS)	Success	Success (gro l
2016-05-27	21:39:00	F9 FT B1023.1	CCAFS LC-40	Thaicom 8	3100	GTO	Thaicom	Success	Success (dr s
2016-05-06	5:21:00	F9 FT B1022	CCAFS LC-40	JCSAT-14	4696	GTO	SKY Perfect JSAT Group	Success	Success (dr s
2016-04-08	20:43:00	F9 FT B1021.1	CCAFS LC-40	SpaceX CRS-8	3136	LEO (ISS)	NASA (CRS)	Success	Success (dr s
2015-12-22	1:29:00	F9 FT B1019	CCAFS LC-40	OG2 Mission 2 11 Orbcomm-OG2 satellites	2034	LEO	Orbcomm	Success	Success (gro l

Section 3

Launch Sites Proximities Analysis

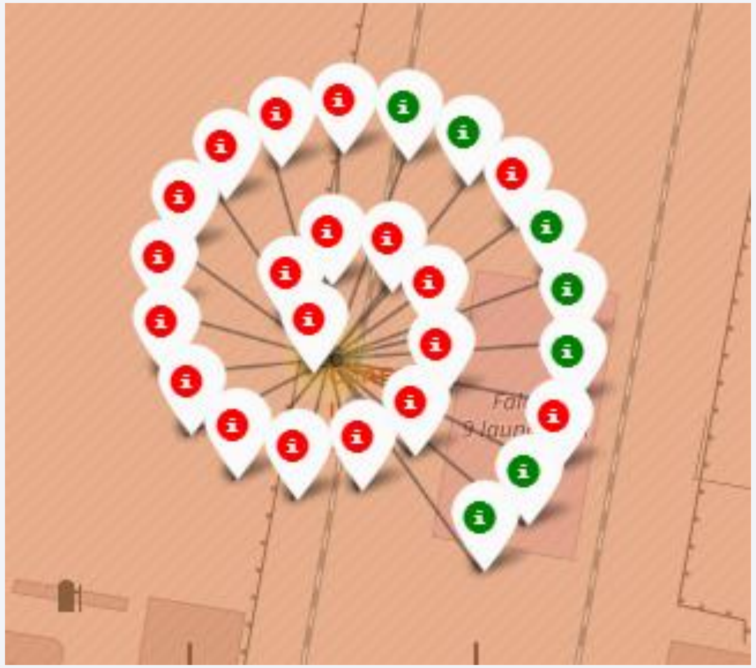
All Launch Sites on a Global Map



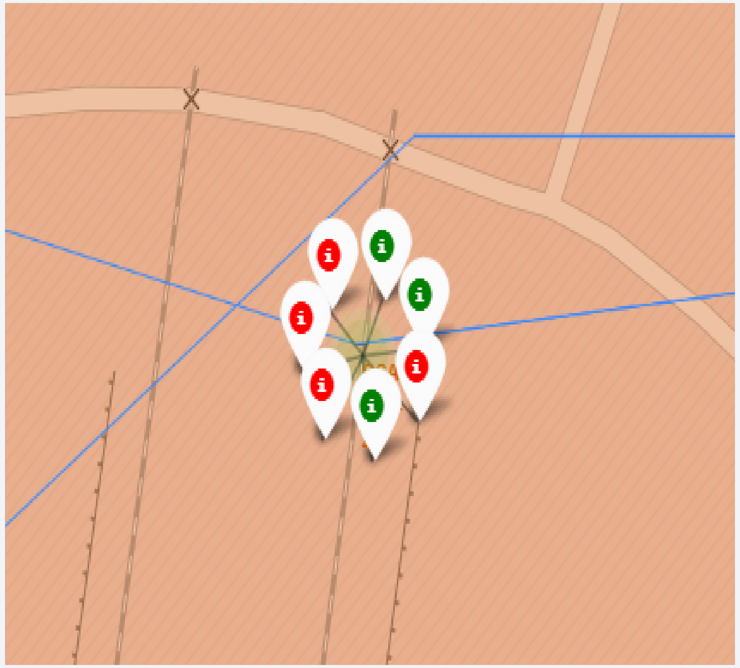
Launch sites are located in USA, California and Florida.

Color-labeled launch outcomes on the map

California Launch Site



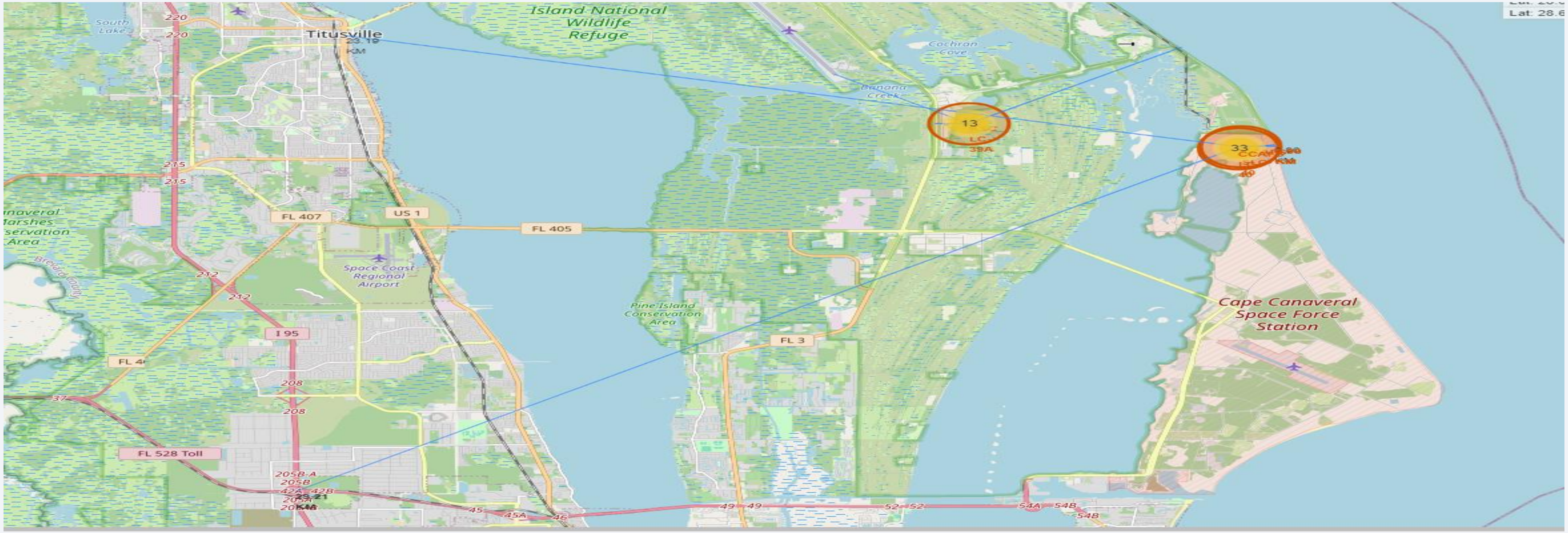
Florida Launch Site



Florida Launch Site

The Green Marker means successful launch and Red Marker means failed launch

Launch Site to the Proximities landmarks



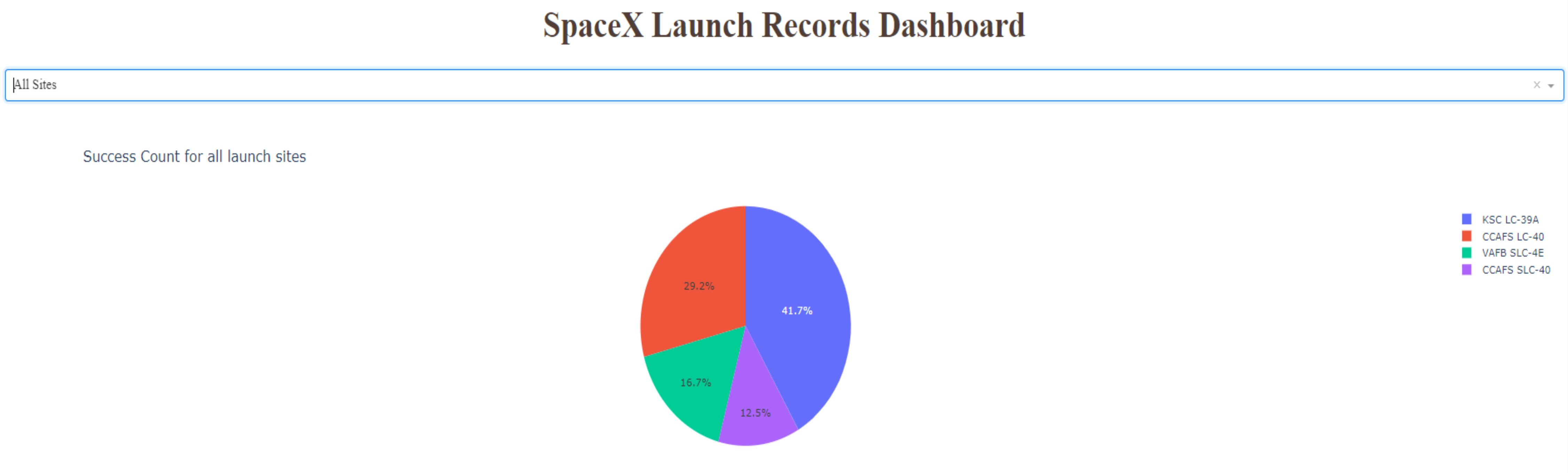
Are launch sites in close proximity to railways?	No
Are launch sites in close proximity to highways?	No
Are launch sites in close proximity to the coastline?	Yes
Do launch sites keep a certain distance away from cities?	Yes

Section 4

Build a Dashboard with Plotly Dash

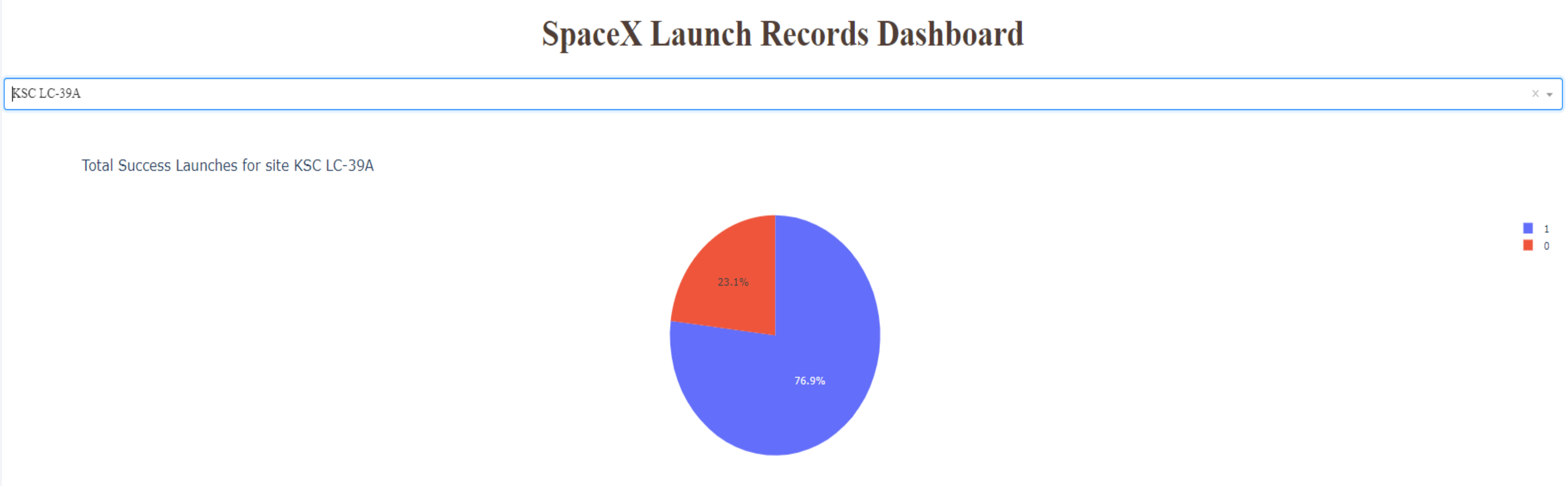
Pie Chart Depicting the Launch Success Counts Across All Sites

Based on the pie chart, the site with the largest number of successful launches is KSC LC-39A, which accounts for 41.7% of the total successful launches across all sites.

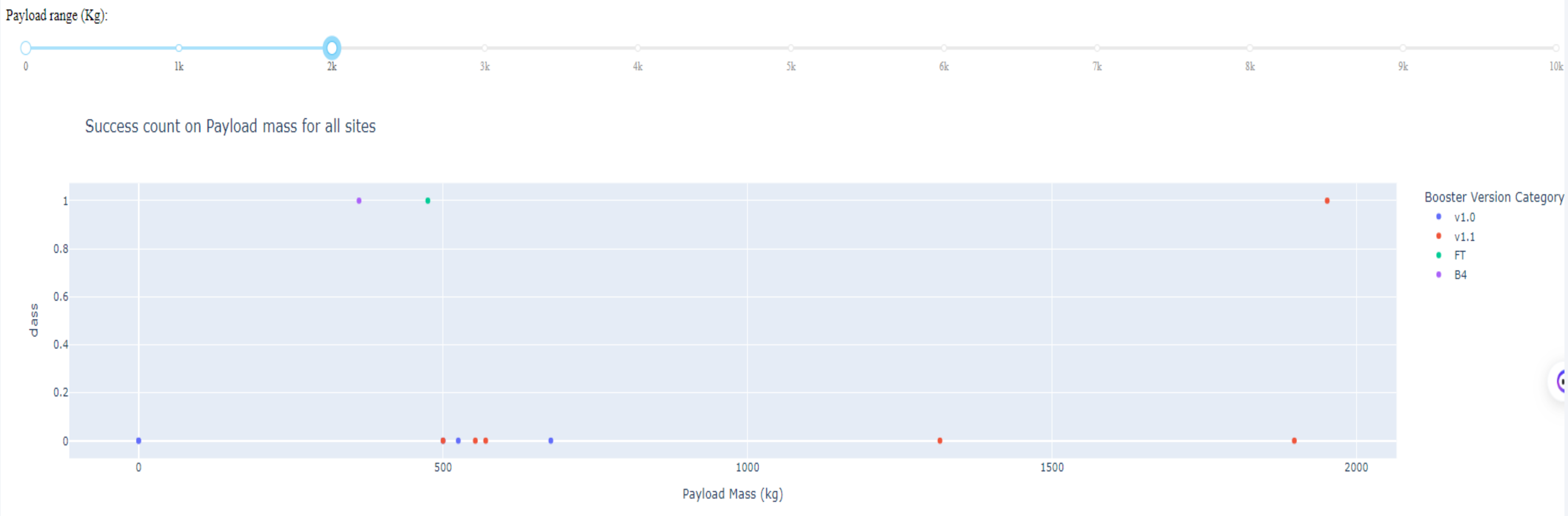


Pie Chart Depicting the Launch Site with the Highest Success Rate

The KSC LC-39A site records a launch success rate of 76.9% and a failure rate of 23.1%.

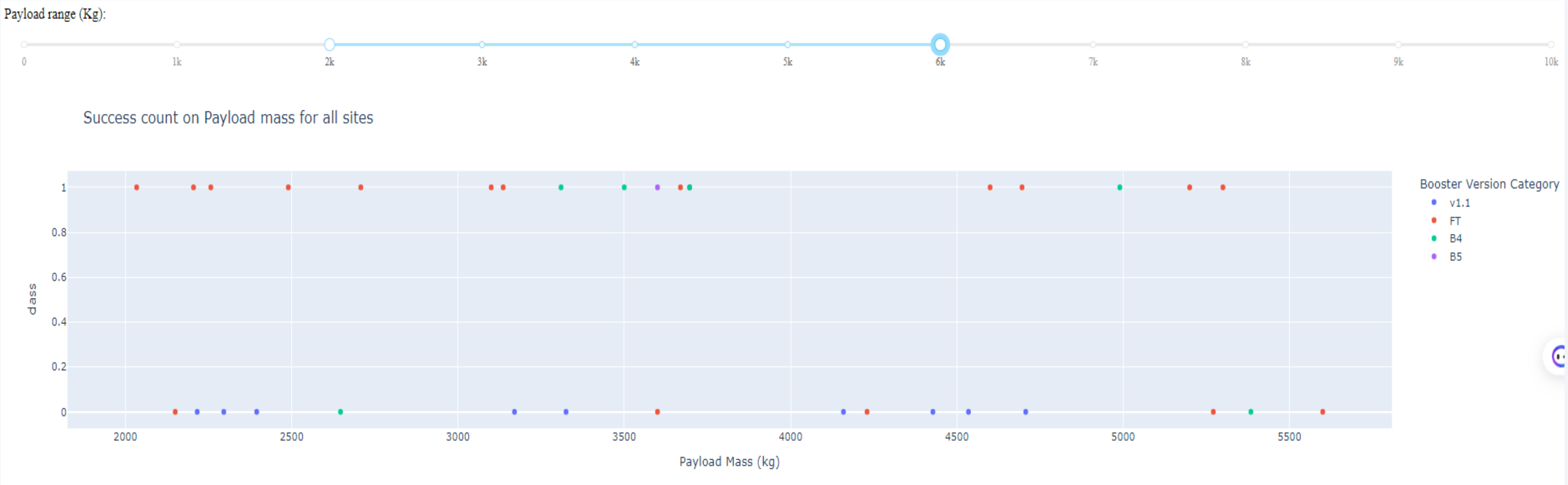


Scatter Plot of Payload vs. Launch Outcome Across All Sites with Payload Selections Using a Range Slider (0 Kg - 2000 Kg)



The payload range 0 kg to just 2,000 kg has scattered successes but also a fair amount of failures, particularly among the earliest booster versions (v1.0 and v1.1).

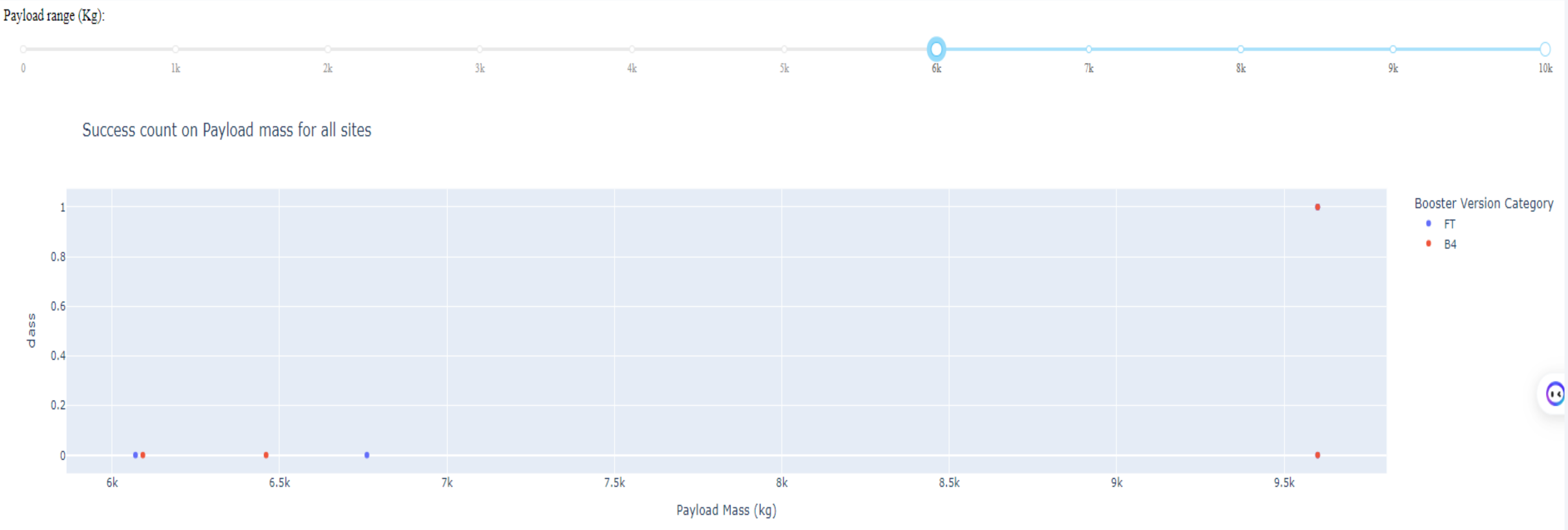
Scatter Plot of Payload vs. Launch Outcome Across All Sites with Payload Selections Using a Range Slider (2000 Kg - 6000 Kg)



From 2,000 kg to 6,000 kg, nearly all launches are successful, across a variety of booster versions, including FT, B4, and B5. This is especially consistent in the 4,000 kg to 6,000 kg range, where there are no visible failures.

Then, the payload range of 2,000 kg to 6,000 kg appears to have the highest success rate, as it shows a consistent pattern of successful launches across various booster versions without any failures indicated in the dataset provided. This range seems to be the most reliable for successful outcomes based on the visualization.

Scatter Plot of Payload vs. Launch Outcome Across All Sites with Payload Selections Using a Range Slider (6000 Kg-10000 Kg)



Beyond 6,000 kg, the number of data points is less, but successes are observed up to 10,000 kg without failures.

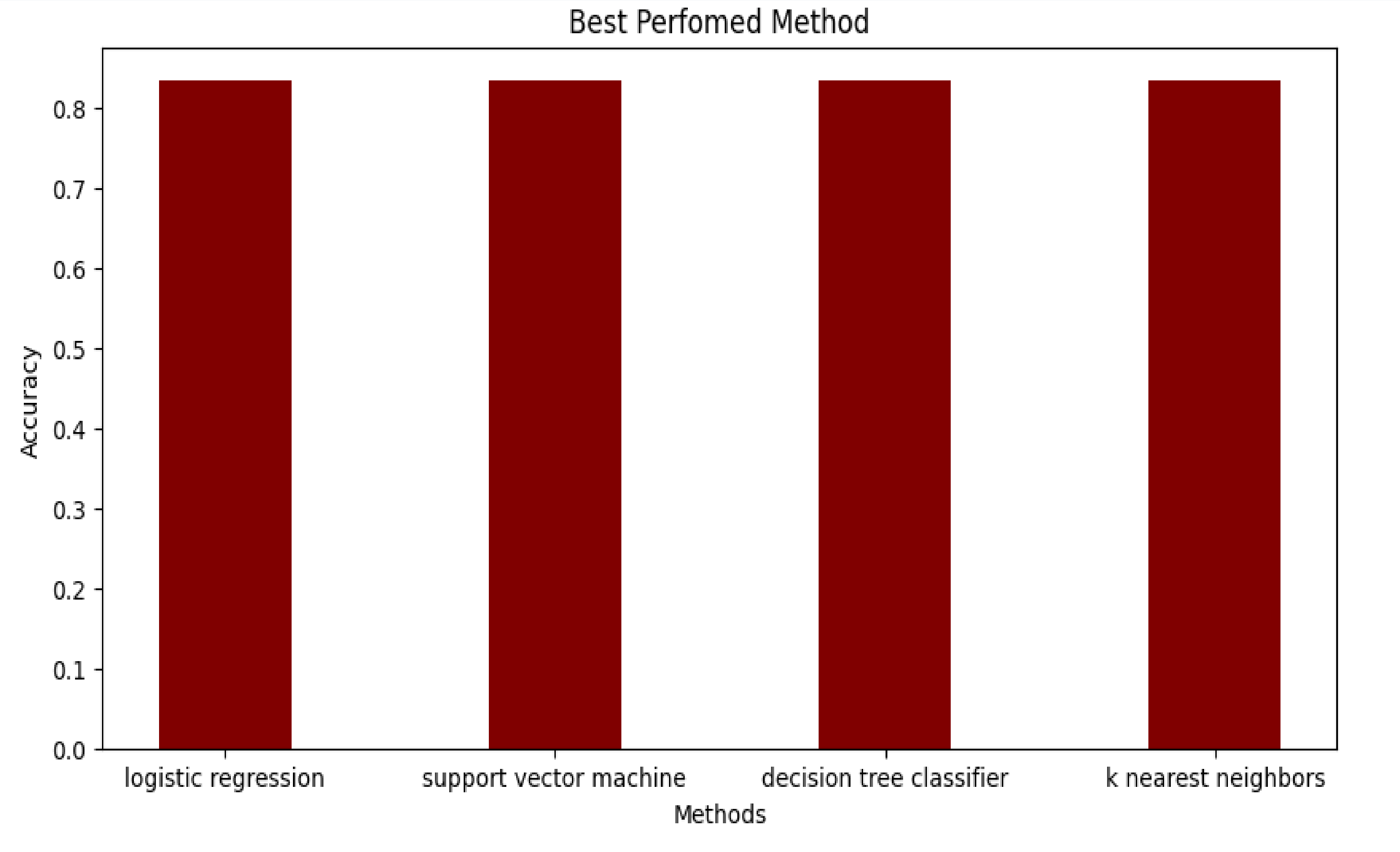
Summary of Observations

1	Which site has the largest successful launches?	KSC LC-39A
2	Which site has the highest launch success rate?	KSC LC-39A
3	Which payload range(s) has the highest launch success rate?	2000Kg-6000 Kg
4	Which payload range(s) has the lowest launch success rate?	4000kg-10000 Kg
5	Which F9 Booster version (v1.0, v1.1, FT, B4, B5, etc.) has the highest launch success rate?	B5 booster version

Section 5

Predictive Analysis (Classification)

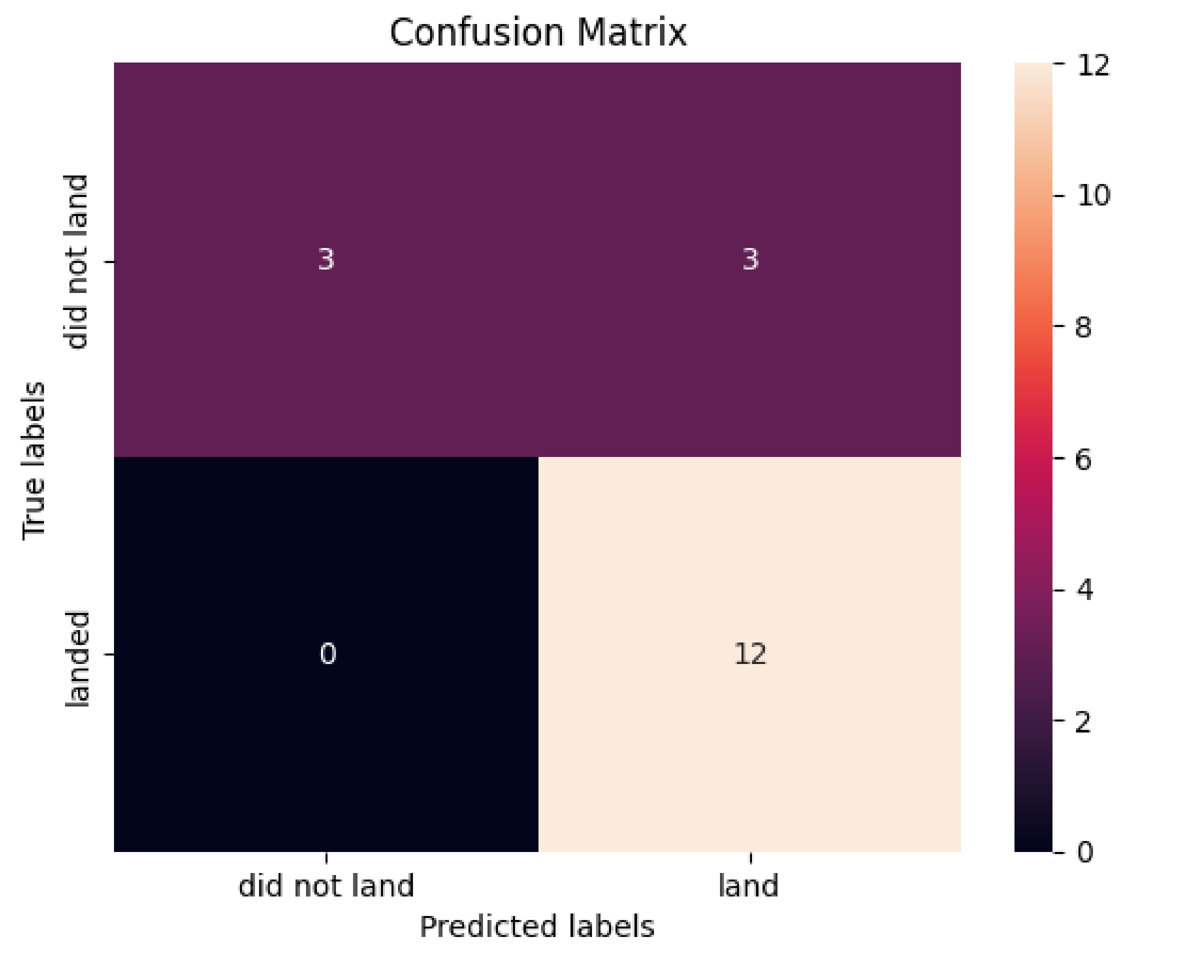
Classification Accuracy



Based on the bar chart, it shows that all four machine learning models—logistic regression, support vector machine, decision tree classifier, and k nearest neighbors—appear to have approximately the same classification accuracy, reaching very close to 0.8 (or 80%). Each model is represented with nearly identical bar heights, indicating that there is no distinguishable difference in classification accuracy among them as shown in the graph.

Hence, no single model among them is depicted as having the highest accuracy; they all perform similarly well according to the data visualized.

Confusion Matrix



All models demonstrated equivalent performance on the test set, as reflected by identical confusion matrices. This indicates a tendency across all models to overestimate the likelihood of successful landings.

Breakdown of Confusion Matrix Graph:

- True Positive (Bottom Right Square, "land"): The model correctly predicted successful landings 12 times. This number represents the landings the model predicted accurately as successful.
- True Negative (Top Left Square, "did not land"): This cell is marked with a '0', indicating there were no instances where the model correctly predicted that a launch would not land successfully (there were no true negative predictions).
- False Positive (Top Right Square, "land"): The model incorrectly predicted successful landings 3 times when the launches did not land successfully. These are false alarms, where the model predicted a landing would succeed, but it did not.
- False Negative (Bottom Left Square, "did not land"): The model does not have any false negatives; it never incorrectly predicted a failure when the landing was actually successful.

Conclusions

- **Data Collection and Processing:** We successfully gathered and refined Falcon 9 launch data, implementing advanced techniques like web scraping and API usage, which provided a comprehensive dataset for analysis.
- **Analytical Insights:** Extensive exploratory data analysis revealed critical insights into factors influencing launch success, highlighting the importance of certain variables such as payload mass and booster version.
- **Machine Learning Predictions:** Our predictive models, including Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K Nearest Neighbors, achieved an accuracy rate of approximately 83.33%. They tended to overpredict successful landings, indicating potential areas for model improvement.
- **Model Evaluation:** The models demonstrated high recall but varied in their precision and specificity, reflecting their ability to detect successful launches but also their limitations in accurately predicting unsuccessful ones.
- **Recommendations for Operational Efficiency:** Analysis suggests optimizing payload ranges and booster versions to enhance the success rates of future launches.

These conclusions underscore the robust capabilities of our analytical framework in driving operational efficiencies and strategic planning for SpaceX's future launch endeavors.

Innovative Insights

- Further research could explore more granular data on environmental conditions and booster wear-and-tear to refine predictions and reduce the models' bias toward successful outcomes.
- To refine the model's predictions and enhance its accuracy, it is recommended to gather more data.

Thank you!